INTERNATIONAL RESEARCH JOURNAL OF ENGINEERING & APPLIED SCIENCES

ISSN: 2322-0821(O) VOLUME 10 ISSUE 3
ISSN: 2394-9910(P) Jul 2022 - Sep 2022

www.irjeas.org

Original Article

Optimal Task Scheduling for Cloud Computing Using Rao Algorithm

*Amrik Singh[1], Puneet Jain[2], Ajay Kumar[3]

[1]*Research Scholar, Electronics and Communication Engineering Dept., Adesh Institute of Engineering & Technology, Faridkot, Punjab, India.* amrik929@gmail.com

[2]*Assistant Professor, Electrical Engineering Dept., Adesh Institute of Engineering & Technology, Faridkot, Punjab, India.* puneetjain988@gmail.com

[3]*Research Scholar, Electronics and Communication Engineering Dept., Thapar Institute of Engineering & Technology, Patiala, Punjab, India.* er.ajay.thapar@gmail.com, https://orcid.org/0000-0002-6061-1626

Corresponding Author – amrik929@gmail.com

https://doi.org/10.55083/irjeas.2022.v10i03003

Received: 18 April 2022; Accepted: 09 July 2022

Abstract: Internet-based provision of computer resources is known as cloud computing. It is possible to utilise data that is controlled by a third party or some other individual at a distant place through cloud computing. Service Level Agreements (SLAs) are used by the majority of Cloud providers to define the services they provide. As part of the SLA, the service provider promises a certain level of quality of the service. Computing and data clouds are two sorts of clouds in a cloud-based system. In cloud technology, task scheduling is critical to ensuring service quality and SLA. One of the most important aspects of cloud computing is a well-organized work schedule. In this article, we have designed an optimal task scheduling method using RAO approach. The RAO algorithm is simple, required number of parameters, and required no tuning of parameters as compared to the other algorithm. Further, a multi-objective function is designed based on RAO algorithm performs the optimal scheduling. The performance evaluation is done by considering number of tasks such as 50,100,200, 300,400, and 500. Further, number of performance metrics are determined for it. The outcomes represents that the presented technique provides lesser values of AWT, ATT, and make span over the existing method.

Keywords: Cloud Computing, Metaheuristic Algorithms, RAO Algorithm, Task Scheduling

1. INTRODUCTION

Cloud computing is a kind of parallel and distributed computing that consists of dynamic, interconnected, and virtualized resources [1]. In cloud technology, resource management has a significant impact on a variety of factors, including price, performance, and effectiveness [2]. Cloud resource management focuses primarily on two issues: allocation of resources and job scheduling. The term "allocation of resources" refers to the process of assigning available resources in accordance with the requirements of individual applications. The goal of task scheduling is to increase profits, optimize usage of resources, and fulfil the needs of the end user.

In the beginning, conventionally-based task scheduling approaches were presented in the literature. Examples of these algorithms are Max-Max, Min-Min, First Come First Served (FCFS), and Priority-based algorithms [3]. These algorithms, on the other hand, do not effectively plan the work and create a huge amount of waiting time. Metaheuristic techniques for job scheduling

have been used to get around this problem [4]. To solve the issue, the metaheuristic algorithms consider the tasks to be a problem and attempt to schedule work based on the objective function. Many optimization algorithms for job scheduling have been proposed in the literature. The GA [5], PSO [6], the FA [7], CSO algorithm [8], and Egyptian vulture optimization algorithm (EVOA) [9] are the most prominent optimization methods. In the GA algorithm, the parents are chosen at random. As a result, a decent offspring generation cannot be achieved if the parents picked are of poor quality. In addition, the present optimization algorithms have a reduced convergence rate to discover the optimum result and local optimization issues [10]. Thus, in order to deal with these issues, the optimization techniques are a hybrid. It is possible to use a combination of methodologies, such as the firefly genetic algorithm [11], the genetic ant optimization algorithm [12], the whale genetic optimization algorithm [13], and hybrid EVOA and firefly algorithm [14]. It is, however, more efficient to use a hybrid approach, but it takes longer to discover the best schedule. Therefore, in this paper, in place of the hybridization approach, we have explored the optimization algorithms that required minimum parameter values and required no parameter tuning as compared to the existing optimization approaches.

The main contribution of this paper is optimal task scheduling in order to decrease the average waiting time. In order to achieve this goal, RAO algorithm is taken under consideration [15]. The proposed method is determined the optimal scheduling based on the objective function. Average waiting time (AWT) is taken as objective function in the proposed method. The simulation evaluation is performed in MATLAB and various parameters such as average waiting time, average turnaround time, make span, and standard deviation are determined for it. The outcomes represents that the presented technique perform superior over other method [9].

The remaining paper is as follows. Section 2 shows the related work. Section 3 explains the preliminaries. Section 4 illustrates the presented method is designed for optimal task scheduling. Section 5 shows the simulation evaluation. In section 6 conclusion of the research article.

2. RELATED WORK

In this section, we have studied the latest methods are designed for optimal task scheduling.

Kaur et al. [9], In this research, we offer a cloud computing task scheduling algorithm that is both efficient and effective. The primary contribution that this work makes is an increase in the convergence rate that may be used to locate the most effective task scheduling. The EVO algorithm is utilised in order to accomplish this goal successfully. In order to locate the best answer, the EVO algorithm executes a series of straightforward tasks, including "striking" it with a rock, "rolling" it with twigs, and "changing the angle." Because of these activities, the exploration rate to discover the ideal solution is significantly higher than that of the already available optimization algorithms. In addition to this, a multi-objective-based fitness function, as opposed to a single-objective fitness function, is intended to discover the best answer possible. In order to validate the suggested method, a number of distinct tasks are taken into consideration, and the size of their respective files is produced at random. In addition, other performance indicators, including such Computation time, AWT, and ATT, are measured.
It has been discovered that the suggested technique performs noticeably better when measured against the existing methods.

Kaur et al. [14], Metaheuristic algorithms are used to solve NP-hard scheduling problems. The goal function-based metaheuristics found the optimal or near-optimal solution. GA , PSO, EVO, as well as FFA are used for work scheduling. This work combines two metaheuristic algorithms to improve optimality. Egyptian vulture optimization delivers an optimal solution. Then, the firefly technique is used to estimate the final global optimum. Parameters like population as well as iterations are also changed to determine the ideal solution. The proposed approach is implemented in MATLAB and its performance metrics are calculated. The new approach is compared to existing methods using various parameters.

Adil et al. [16], This study investigates Meta-Heuristic for Hadoop cloud work scheduling. Hadoop is an open implementation of the Map Reduce framework used to handle computationally expensive jobs on a multi-node cluster. The scheduling algorithm determines the order and node of job execution to create an efficient schedule. A scheduling algorithm employs execution time, task arrival order, and data location to produce the best consequently result. PSO is used to determine task execution schedules and compared to GA, BF, FIFO, as well as DSP algorithms. This study proves PSO's superiority over other algorithms for cloud work scheduling.

3. PRELIMINARIES

In the section, RAO algorithm is explained to understand the proposed method is designed for optimal task scheduling.

Ravipudi Venkata Rao (2020) proposed Rao algorithms, metaphor-free optimization [15, 17]. Rao Algorithms use best as well as worst outcomes in each round and random interactions to quickly find an optimal solution. Figure 1 shows RAO's flowchart. The RAO based on the determination, population size, design factors, and completion condition. Then, depending on the research function, each population's fitness is evaluated to find the best as well as worst solution. Customize solutions depending on best, worst, and random events, as shown. If the modified solution is better than the best, take the best solution. The whole operation is repeated until termination criterion is not found. In the last, optimal solution is reported. In the proposed method, optimal starting pixel in the cover image is determined using the RAO algorithm.

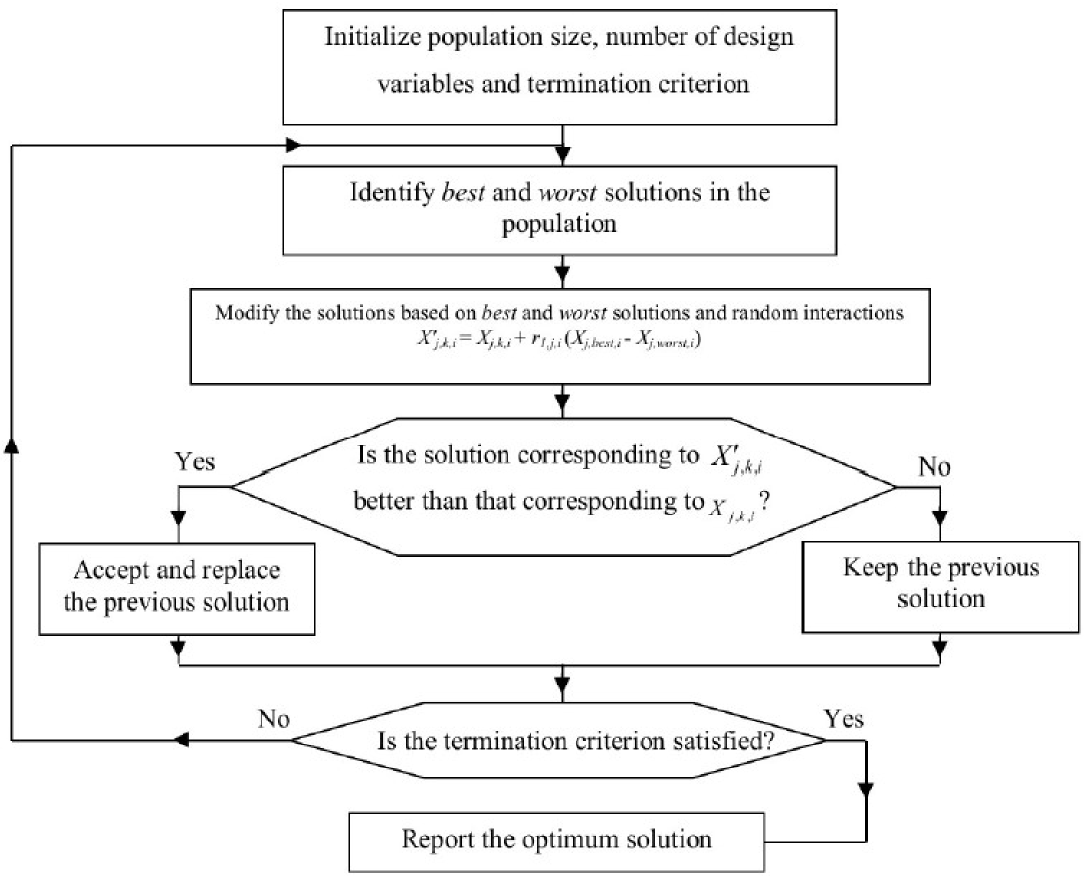

Figure 1- Flowchart of RAO Algorithm

4. PROPOSED METHOD

The presented technique is based on RAO approach for optimal task scheduling. The flowchart of the proposed method is designed for optimal task scheduling shown in Figure 2.

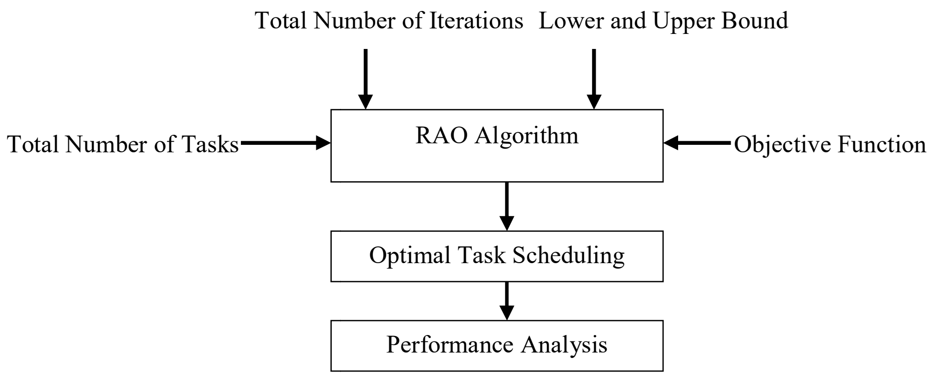

Figure 2- Flowchart of Optimal Task Scheduling using RAO Algorithm

In the proposed method, initially, total number of tasks, iterations, lower and upper bound, and objective function are input to the RAO algorithm for optimal task scheduling. The steps are taken in the proposed method for optimal task scheduling is explained below.

1. Initially, execution time is generated in the lower and upper bound range for total number of tasks.

2. The total number of populations is defined and each population is randomly permutated to generate different task scheduling.

3. The fitness evaluation of each population is done using objective function. In the proposed method, average waiting time (AWT) is taken as objective function.

4. Based on objective function, best and worst population is determined.

5. Next, new population is generated based on the best, worst, and random interaction parameters.

6. The new population's fitness is matched to the best population, then updated if necessary.

7. The whole operation is iterated for fixed number of iterations.

The presented program's results are analyzed utilizing average waiting time, average turnaround time, and computation time.

5. SIMULATION RESULTS

The presented technique is simulated in MATLAB. The setup configuration for the RAO algorithm is shown in Table 1.

Table 1- Setup Configuration for the RAO Algorithm

Parameters	Value
Total Population	20
Total Iterations	500
File Size	32-64MB
Total Number of Tasks	50
Objective Function	AWT

Table 2 explains the performance parameters are determined for the proposed method [9, 14].

Table 2- Performance Parameters

Parameters	Equation
Average Waiting Time (AWT)	$AWT = \frac{\sum_{i=1}^{N-1} C_i}{N}$
Average Turnaround Time (ATT)	$ATT = \frac{\sum_{i=1}^{N} C_i}{N}$
Makespan	Makespan=$max_{i \in task}\{C_i\}$

Next, the simulation evaluation of the presented technique using various parameters.

Table 3 shows the simulation evaluation of the presented technique based on AWT parameter.

Figure 3 shows that the proposed method lesser AWT time over the original AWT.

Table 3- Average Waiting Time

Tasks	Original AWT	Proposed Method AWT
50	1142.70	1133.10
100	2449.40	2401.60
200	4674.40	4639.50
300	7221.90	7115.00
400	9421.70	9416.30
500	11863.00	11798.00

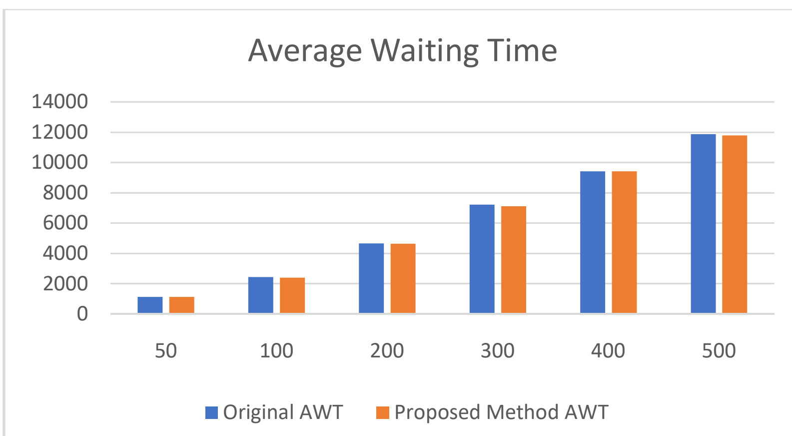

Figure 3- Comparative Analysis of Original and Proposed Method AWT

Table 4 shows the simulation evaluation of the presented technique based on ATT parameter.

Figure 4 represents that the presented technique lesser ATT time over the original ATT.

Table 4- Average Turnaround Time

Tasks	Original ATT	Proposed Method ATT
50	1190.30	1180.70
100	2498.80	2451.00
200	4721.50	4686.60
300	7270.00	7163.00
400	9469.40	9463.90
500	11910.00	11846.00

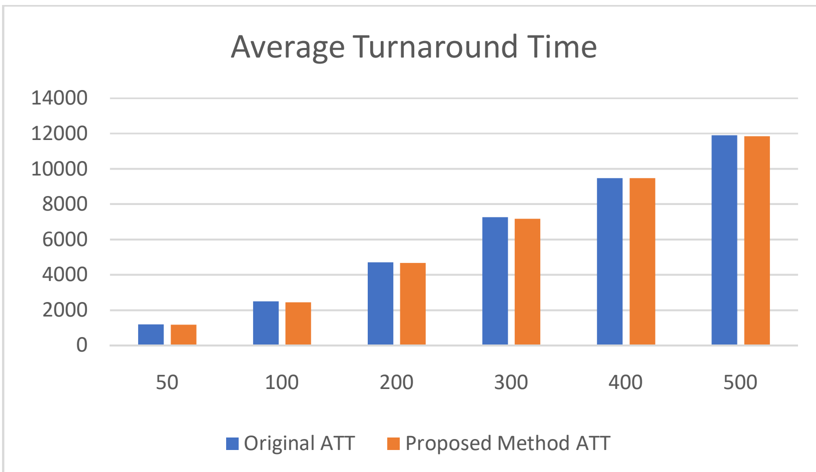

Figure 4- Comparative Analysis of Original and Proposed Method AWT

Table 5 represents the simulation evaluation of the presented technique based on makespan parameter.

Figure 5 shows that the proposed method lesser makespan time over the original makespan.

Table 5- Makespan

Tasks	Original Makespan	Proposed Method Makespan
50	59515	59035
100	249882	245102
200	944304	937322
300	2180986	2148910
400	3787742	3785558
500	5955112	5922942

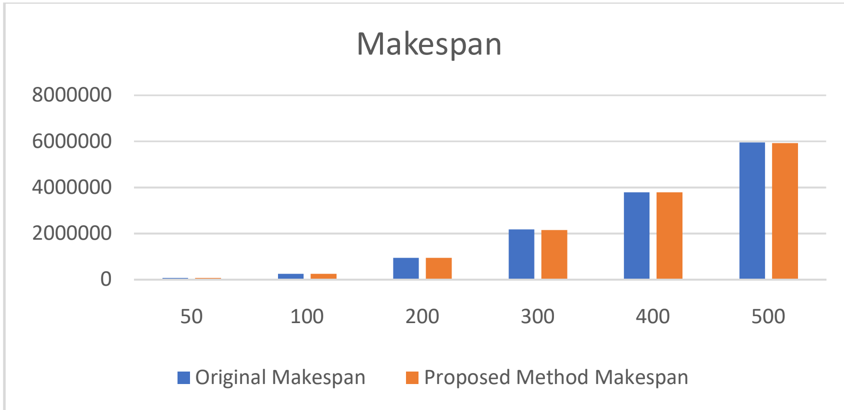

Figure 5- Comparative Analysis between Original and Proposed Method Makespan

5.1. Comparative Analysis

In this part, suggested and current approaches are compared. To compare suggested and existing methods, the designed code is simulated independently 50 times and average value of AWT, ATT, and makespan is determined. Table 6 compares the obtained from the presented AWT-based methods. Figure 6 indicates that the presented technique lowers AWT than the conventional method.

Table 6- Comparative analysis of the presented technique with the existing technique depends on AWT

Tasks	EVOA [9]	Proposed Method
50	1275.6	1145.30
100	2328.9	2330.50
200	4932.5	4718.00
300	7166.3	7102.60
400	9510.9	9485.80
500	12774	11891.00

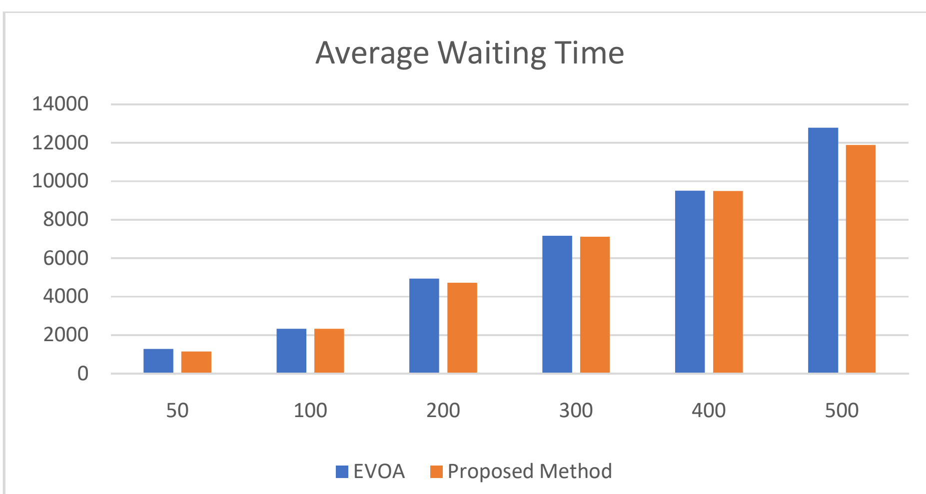

Figure 6- Comparative Analysis based on AWT

Table 7 shows the comparative analysis of the presented technique with the existing method based on ATT parameter. The result shows that the presented technique achieves lower AWT as compared with other method, as shown in Figure 7.

Table 7- Comparative analysis of the presented technique with the existing technique depends on ATT

Tasks	EVOA [9]	Proposed Method
50	1334.2	1193.30
100	2381	2378.50
200	4985.4	4766.00
300	7217.5	7150.60
400	9560.7	9533.80
500	12827	11939.00

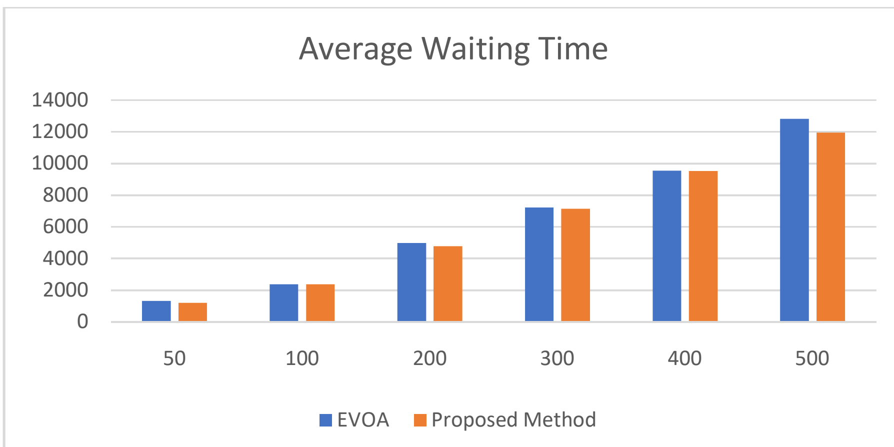

Figure 7- Comparative Analysis based on ATT

Table 8 shows the comparative analysis of the presented technique with the existing method based on AWT parameter. The result shows that the presented technique obtains lower AWT as compared to the existing method, as shown in Figure 8.

Table 8- Comparative analysis of the proposed method with the existing technique depends on Makespan

Tasks	EVOA [9]	Proposed Method
50	66711	65711
100	238096	237096
200	997077	996077
300	2165239	2164239
400	382467	381467
500	6413306	5412306

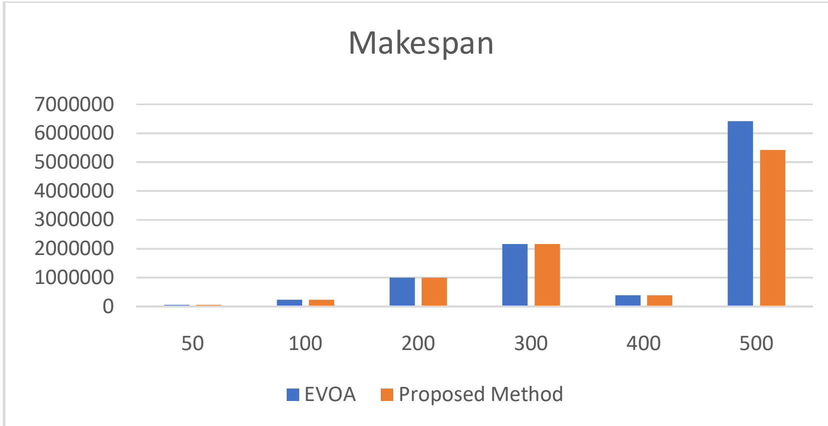

Figure 8- Comparative Analysis based on Makespan Parameter

6. CONCLUSION

In this paper, an optimal task scheduling method is designed using swarm intelligence algorithm. In the swarm intelligence algorithm, RAO algorithm is taken under consideration. The RAO algorithm is simple, required minimum number of parameters, and no tuning of parameters are required. The RAO algorithm searches the optimal task scheduling depends on the objective function. The average waiting time is taken as objective function in the proposed method. The simulation evaluation is done by considering the number of tasks and outcomes represents that the presented technique provides lesser values of AWT, ATT, and makespan as compared with other technique EVOA based method [9].

REFERENCES

[1] Suri, P.K. and Rani, S., 2017, May. Design of task scheduling model for cloud applications in multi cloud environment. In *International Conference on Information, Communication and Computing Technology* (pp. 11-24). Springer, Singapore.

[2] Buyya, R., Yeo, C.S., Venugopal, S., Broberg, J. and Brandic, I., 2009. Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility. *Future Generation computer systems*, *25*(6), pp.599-616.

[3] Singh, P., Dutta, M., & Aggarwal, N. (2017). A review of task scheduling based on meta-heuristics approach in cloud computing. *Knowledge and Information Systems*, *52*(1), 1-51.

[4] Adil, S. H., Raza, K., Ahmed, U., Ali, S. S. A., &Hashmani, M. (2015, December). Cloud task scheduling using nature inspired meta-heuristic algorithm. In *2015 International Conference on Open Source Systems & Technologies (ICOSST)* (pp. 158-164). IEEE.

[5] Omara, F. A., & Arafa, M. M. (2009). Genetic algorithms for task scheduling problem. In *Foundations of Computational Intelligence Volume 3* (pp. 479-507). Springer, Berlin, Heidelberg.

[6] Alsaidy, S. A., Abbood, A. D., & Sahib, M. A. (2020). Heuristic initialization of PSO task scheduling algorithm in cloud computing. *Journal of King Saud University-Computer and Information Sciences*.

[7] Saleh, I. A., Alsaif, O. I., Muhamed, S. A., & Essa, E. I. (2019, September). Task scheduling for cloud computing based on firefly algorithm. In *Journal of Physics: Conference Series* (Vol. 1294, No. 4, p. 042004). IOP Publishing.

[8] Singh, S., & Singh, K. P. (2015). Cuckoo search optimization for job shop scheduling problem. In *Proceedings of Fourth International Conference on Soft Computing for Problem Solving* (pp. 99-111). Springer, New Delhi.

[9] Kaur, G. and Sharma, A., (2021). Optimal Task Scheduling using Egyptian Vulture Optimization Algorithm. *Design Engineering*, pp.12542-12552.

[10] Kaur, A., Pal, S. K., & Singh, A. P. (2020). Hybridization of Chaos and Flower Pollination Algorithm over K-Means for data clustering. *Applied Soft Computing*, *97*, 105523.

[11] Rajagopalan, A., Modale, D. R., &Senthilkumar, R. (2020). Optimal scheduling of tasks in cloud computing using hybrid firefly-genetic algorithm. In *Advances in decision sciences, image processing, security and computer vision* (pp. 678-687). Springer, Cham.

[12] Kumar, A. S., & Venkatesan, M. (2019). Multi-objective task scheduling using hybrid genetic-ant colony optimization algorithm in cloud environment. *Wireless Personal Communications*, *107*(4), 1835-1848.

[13] Natesan, G., &Chokkalingam, A. (2020). Multi-objective task scheduling using hybrid whale genetic optimization algorithm in heterogeneous computing

environment. *Wireless Personal Communications*, *110*(4), 1887-1913.

[14] Kaur, G. and Sharma, A., (2022). Hybrid Egyptian Vulture Optimization Algorithm with Firefly Algorithm for Optimal Task Scheduling In Cloud Environment. *Journal of East China University of Science and Technology*, *65*(2), pp.57-66.

[15] Suyanto, S., Wibowo, A. T., Al Faraby, S., Saadah, S., &Rismala, R. (2021). Evolutionary rao algorithm. *Journal of Computational Science*, *53*, 101368.

[16] Adil, S. H., Raza, K., Ahmed, U., Ali, S. S. A., &Hashmani, M. (2015, December). Cloud task scheduling using nature inspired meta-heuristic algorithm. In *2015 International Conference on Open Source Systems & Technologies (ICOSST)* (pp. 158-164). IEEE.

[17] Rao, R., (2020). Rao algorithms: Three metaphor-less simple algorithms for solving optimization problems. *International Journal of Industrial Engineering Computations*, *11*(1), pp.107-130.

Conflict of Interest Statement: *The authors declare that there is no conflict of interest regarding the publication of this paper.*